UCRL- 92588
PREPRINT

# A USER INTERFACE ON NETWORKED WORKSTATIONS FOR MFTF PLASMA DIAGNOSTIC INSTRUMENTS

V. L. Renbarger
T. R. Balch

This paper was prepared for submittal to 11th
Symposium on Fusion Engineering
Austin, Texas
November 18-22, 1985

October 2, 1985

Lawrence Livermore National Laboratory

# A USER INTERFACE ON NETWORKED WORKSTATIONS FOR MFTF-B PLASMA DIAGNOSTIC INSTRUMENTS

T. Richard Balch and V.L. Renbarger
Lawrence Livermore National Laboratory
P.O. Box 5511, L-535
Livermore, CA 94550

## Abstract

A network of Sun-2/170 workstations is used to provide an interface to the MFTF-B Plasma Diagnostics System at Lawrence Livermore National Laboratory. The Plasma Diagnostics System (PDS) is responsible for control of MFTF-B plasma diagnostic instrumentation. An EtherNet Local Area Network links the workstations to a central multiprocessing system which furnishes data processing, data storage and control services for PDS. These workstations permit a physicist to command data acquisition, data processing, instrument control, and display of results.

The interface is implemented as a metaphorical desktop, which helps the operator form a mental model of how the system works. As on a real desktop, functions are provided by sheets of paper (windows on a CRT screen) called worksheets. The worksheets may be invoked by pop-up menus and may be manipulated with a mouse. These worksheets are actually tasks that communicate with other tasks running in the central computer system. By making entries in the appropriate worksheet, a physicist may specify data acquisition or processing, control a diagnostic, or view a result.

## Introduction

MFTF-B (Mirror Fusion Test Facility), located at Lawrence Livermore National Laboratory, is a large-scale facility for tandem-mirror fusion energy research. A large integrated network of computers provide control and diagnostic functions to operators and physicists [1, 2].

The Plasma Diagnostics System (PDS) will ultimately manage 22 separate diagnostic systems to monitor operational parameters in the reactor vessel before, during, and after experimental physics shots or sequences. Data collected from the diagnostic instruments will be processed, archived, and then displayed to physicists by PDS. This information will be used to manage the experiment and ongoing vessel operations.

In [13] the diagnostics user interface using an 'electronic desktop' metaphor was introduced. This paper will discuss the further design and actual implementation of the PDS user interface using the desktop metaphor. While similar in appearance to commercially available systems, our interface allows physicists to actively control and monitor remote diagnostic instruments, schedule data acquisition and processing, and view current or archived processing results. The PDS desktop interface provides a logically consistent framework from which physicists will be able to effectively manage the large number of widely heterogeneous diagnostic instrument systems.

This paper discusses our user interface to the MFTF-B diagnostic system--the environment it runs in (computer and non-computer), how it was implemented, and how specific functions are provided to the operator.

## The Environment

The central computer system for MFTF-B is the Supervisory Control and Diagnostics System (SCDS). It consists of nine Perkin-Elmer 3200 series mini-computers networked through a shared memory [14]. There are also a number of associated LSI-11 local control computers, graphics workstations, and control consoles. Detailed descriptions of MFTF-B computer hardware can be found in [1,2,10,11].

Two of the nine minicomputers are dedicated to diagnostic control and data processing. It was anticipated that these two Diagnostic Data Processors (DDPs) would be heavily loaded by PDS data processing requirements, and so the user interface and graphics display tasks have been off-loaded onto intelligent workstations. We currently have three Sun Microsystems 2/170 workstations linked to one of the DDPs via communications gateways utilizing an EtherNet link. The other DDP is being used for code development and is not currently available for diagnostic data processing. Figure 1 illustrates the current configuration.

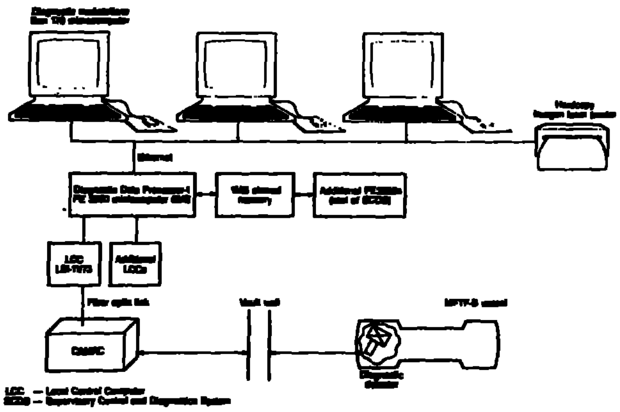

Figure 1. Hardware configuration

Each Sun-2/170 workstation is a MC68010-based, 32-bit microcomputer with a high-resolution bit-mapped graphics display, an optical mouse pointing device, and an EtherNet interface. One of the workstations is equipped with a 400 megabyte disk and acts as a network file server. The workstations run Berkeley version 4.2 Unix, with networking and graphics extensions. An Imagen laser printer is also available to the workstations on the network.

Within this system environment, PDS has been designed and implemented to meet diagnostic requirements for MFTF-B operations. Physics shots may occur as frequently as every five minutes, and up to eight megabytes of raw diagnostic data may be acquired during each shot. Our system must have the capability of moving the raw data into archival units, processing the data, and displaying a subset of the results within the five-minute shot cycle. The physicists will use these inter-shot results to adjust physics parameters for subsequent shots.

PDS has to be able to control diagnostics, specify data acquisition and data processing, and to view results [13]. Physicists must not only be able to view a subset of results between shots, but they must also be allowed to display old results. Further, physicists can schedule immediate processing of archived data (which runs at a lower priority) and view the subsequent results.

## The Diagnostic Desktop Metaphor

Several successful commercial packages have emerged that have a central theme or metaphor which enables operators to utilize the system more easily and effectively. Xerox pioneered this idea with the Star Information System [3], while a more recent example is the Apple Macintosh [4]. These systems use the display screen to represent an office desktop, complete with sheets of paper, calculators, and file folders. The sheets of paper, or "windows"[1], may actually represent more complex objects than would appear on a conventional desktop. Examples are text editors, chart generators, and mathematical spreadsheets.

We chose to use a similar desktop metaphor for the PDS user interface. There is a basic desktop layout which provides the operator with diagnostic information and menu choices. Using the mouse, the operator can cause new windows to appear on the screen by pointing at these menu choices and pushing a button. These windows (called 'worksheets' in our application), allow for greatly expanded capabilities than those found in commercial products. Using PDS worksheets operators can control and monitor diagnostic hardware, run complex control sequences, schedule processing and results displays, examine old results, or process archived data. An example desktop is shown in Figure 2.

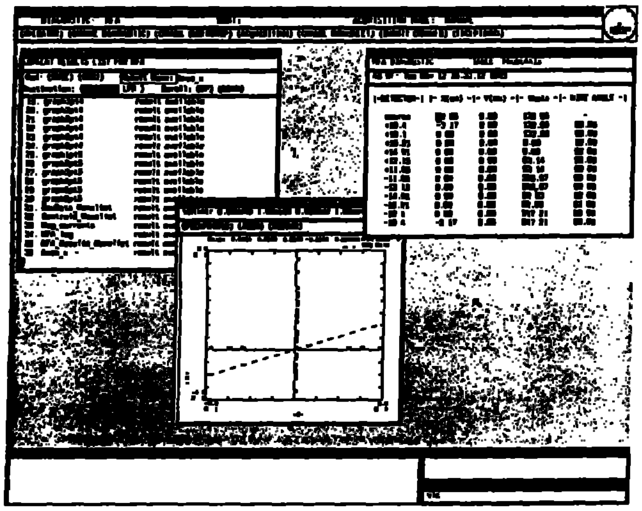

Figure 2. Desktop with Control Table and Results Worksheets

This desktop displays worksheets for doing control and viewing results. In this example we are trying to illustrate how the PDS desktop metaphor, in conjunction with a mouse, provides a very powerful but consistent and easy-to-use interface to the Plasma Diagnostics System. The desktop metaphor is powerful because operators can manage the diagnostics system, view results, etc. by simply making entries or picking selections in worksheets from our remote workstations. The desktop metaphor is consistent and easy-to-use because everything is implemented using worksheets and pop-up menus, and all user inputs into the system come from the keyboard or mouse.

## Implementation of the Desktop

Some general purpose utilities were built based on Sun's library of window management routines [7]. Sun's routines give the C [6] programmer access to the bit-mapped graphics display, including low-level pop-up menu generation and windowing capabilities.

A general-purpose hierarchical menu system, which takes advantage of Unix [5] multitasking and task forking capabilities, was also created. The system presents the operator with a hierarchical series of menus through which he or she can select and create worksheets. The menu system is file-driven, so it is easily modified by editing a text file; no code modification is required to implement a new menu set. Other tools include a standard worksheet program and several user interface routines.

There was also a need to provide desktop worksheets with general information (information global to the desktop environment) needed to administrate the system. This included things such as operator login status, the 'default diagnostic' (diagnostic of current interest), and diagnostics 'owned' by the operator (a list of diagnostics for which the operator has been granted exclusive control privileges). To provide this information, a 'global information server' task was implemented. This task runs as a background process on each workstation. The global server enables PDS to implement its control policies--allowing multiple operators to view control worksheets on different workstations, but only allowing one authorized operator at a time to actually control a diagnostic. The workstation tasks use subroutine calls to retrieve information from the global server.

## The PDS Desktop

The basic PDS desktop consists of an information banner, a command menu, a current operator window, an owned diagnostics window, an exceptions window, and a clock (see Figure 2). The name of the current operator (logged-in user) is displayed in the Current Operator window and the diagnostics that the operator controls ('owns') are displayed in the Owned Diagnostics window. The Exception window displays notifications of exceptional conditions (alarms) related to the diagnostics system.

The command menu presents functional-level choices to the operator. Using a hierarchical menu system, a command menu item pick results in the next level of menu selections for that command item. An operator may need to traverse down through several levels of increasingly specific menu items to arrive at a worksheet or other desired operation.

The highest level choices provide the following functions:

Operator -

1) Login by entering name and password.
2) Logout.
3) Display all current operators and the workstations that they are logged in on.
4) Display all possible operators and the access rights that they have (access rights regulate which diagnostics an operator can own and the functions he or she can legally perform).

[1] A window is a rectangular subset of the CRT screen. Typically a screen may contain multiple windows, each interacting with the operator independently of the others.

**Choose Diagnostic -**

The operator selects the diagnostic of interest from a list of the diagnostics currently implemented on PDS. The selected diagnostic is the operator's 'default diagnostic' and subsequent menu selections which are diagnostic-related will automatically use this diagnostic.

**Choose Ownership -**

1) Request ownership (sole control privileges) of an available diagnostic.
2) Release ownership of a diagnostic.
3) Display the current owners of all diagnostics.

**Acquisition -**

1) Change data acquisition mode (automatic or manual). The acquisition mode determines whether a diagnostic will be included automatically in the next experimental physics shot. 'Automatic mode' places the diagnostic into the shot participation list--any data acquisition and data processing which has been previously scheduled will occur each subsequent physics shot until the diagnostic mode is changed to manual mode.
2) Perform manual acquisition control functions. This allows operators to manually trigger, schedule, or abort data acquisition for diagnostic instruments.

**Choose Worksheet -**

The choices that appear in the pop-up menu allow the operator to bring up different worksheets on the desktop (control table worksheets, processing results worksheets, etc.). The lower levels of worksheet menus are based on the default diagnostic (i.e. different diagnostics may have different worksheet requirements).

**Submit Changes -**

The menu selections for this command item allow the operator to simultaneously update all pending control, acquisition, or processing worksheet entries. This simply allows operators to make a number of changes on several worksheets and then send updates all at once to the DDP. This avoids having to send updates for each individual worksheet.

**Exceptions -**

Functions include:

1) Displaying exceptions.
2) Displaying help information for exceptions.
3) Setting exception limits for a diagnostic (e.g. alarm values).
4) Override a particular exception.

Figure 3 illustrates the steps an operator performs to call up a specific worksheet.

## Diagnostic Control

Control Diagnostic Equipment (CDE), a task running on the DDP, manages the control functions for each diagnostic. The interface to CDE on the workstation consists of control table worksheets. These worksheets resemble commercially available spreadsheets, although they provide far greater capabilities. They allow operators to control diagnostic hardware, monitor hardware status, and perform complex communication and control sequences. (See [8] for a complete description of control tables.)

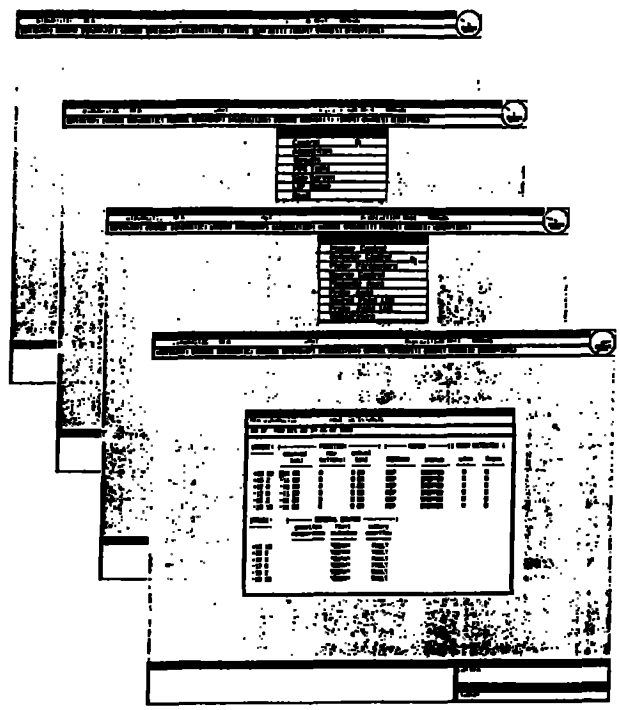

Figure 3. Calling up a Control Table Worksheet

The control table worksheet is the operator's primary source of status and configuration information for a diagnostic. In addition to providing status information, the control table worksheet allows the physicist to actually control diagnostic hardware by changing values in control table fields. Each control table worksheet is designed to reflect the physical design of a portion of a diagnostic. Typically there may be numerous control tables for a diagnostic. For example, the Magnetic Field Alignment (MFA) diagnostic requires nine control tables [9]. The tabular format, in conjunction with the windowing system, allows great flexibility in logically grouping related status, control, or configuration fields.

The following examples illustrate the flexibility in layout and functions available for control table worksheets. The Source Control table worksheet (Figure 4) provides fields for immediate control of a diagnostic for sequencing or positioning an instrument.

The 'Source Point List' control table (Figure 5) allows the operator to describe actions the diagnostic should take during a data-gathering sequence.

To implement the PDS control policy, control table worksheet tasks query the global server on the workstation for information about the operator. Only authorized operators are allowed to change table values. The updated field values are then forwarded to the CDE task on the DDP machine.

When CDE is informed of a control table change from the workstation, it updates its internal tables and sends updates to all current control table displays on other workstations. Some control table fields have strings of an interpreted command language associated with them [8] which cause calculations to be performed or control commands issued. Control commands are sent to a control and data acquisition program called PLEX [10, 11] on the local control computers. PLEX in turn communicates with the diagnostic instruments (see Figure 6).

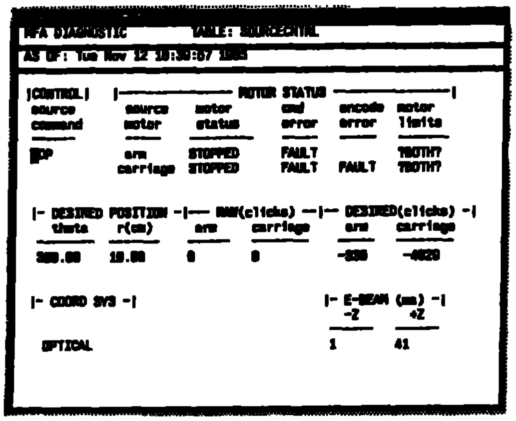


Figure 4. MFA Source Control Table

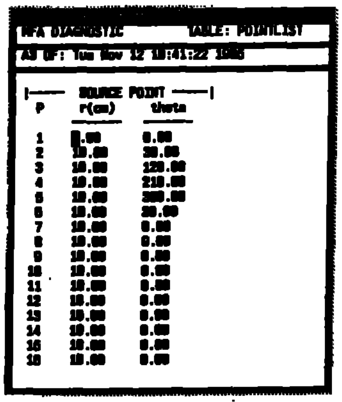


Figure 5. MFA Source Point List Table

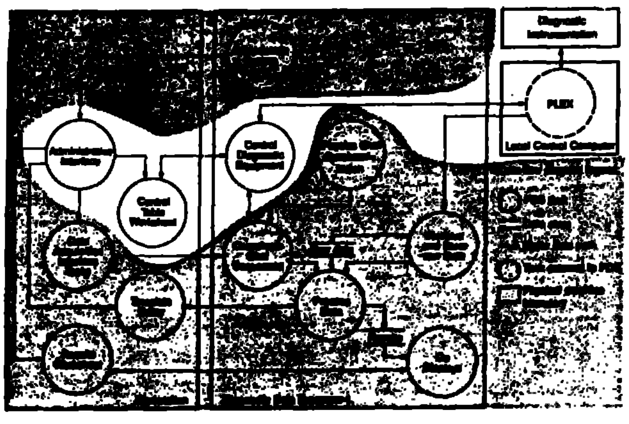


Figure 6. Data flow for diagnostic control

When the status of a diagnostic instrument changes (i.e. movement, temperature, error conditions), PLEX informs CDE. CDE updates its internal tables and notifies the appropriate worksheets on the workstations of the changes.

## Data Acquisition

There are two aspects of data acquisition: what data vectors to acquire and when to acquire them. The actual specification for data vector acquisition is done via 'acquisition table' worksheets. These worksheets are very similar in appearance to control table worksheets and are chosen from the 'Choose Worksheets' command menu. Requesting an acquisition table worksheet results in a request to CDE on the DDP for the list of available data vectors for the current default diagnostic. The acquisition table worksheet displays the list of data vector names and the operator can then select which data vectors he would like acquired. When the selections have been completed, they are sent to CDE. CDE uses these selections to schedule data acquisition from the diagnostic instruments. The operator is responsible for making sure that all data vectors required by processing are selected for data acquisition.

A diagnostic runs in either automatic or manual data acquisition mode. Specifying a diagnostic to run in automatic mode means that the diagnostic is put into the shot participation list and everytime a physics shot occurs, data is gathered for that particular diagnostic. If a diagnostic is in manual mode, the operator has the option of asking for data to be collected immediately or for it to be collected on the next timing signal (physics shot or conditioning shot). For whichever of these two manual options the operator picks, data will only be acquired once--unlike automatic mode where the data would be acquired on every shot as long as the diagnostic remained in the shot participation list. When a diagnostic is in manual mode, the operator may also abort the ongoing data collection for that diagnostic. By picking 'Acquisition' in the desktop command menu, the operator can specify which mode a diagnostic will run in. If the diagnostic is in manual mode, he can also choose any of the above mentioned manual options.

## Specification of Processing

Processing is specified by operators in a high level language called 'Yak' [12]. From a workstation 'processing command' worksheet called the Template Editor, the operator schedules data processing using Yak commands. The Template Editor functions much like a text editor, and provides language primitives to define processes, inputs, and outputs. Yak also allows for processes to be chained (one process depending on output from a previous process). The completed list of processing commands is called a 'processing template'.

When a processing template has been completed, it is sent over to a process-scheduling task on the DDP ('Process Data'). This task stores the processing template, and uses it to schedule and run processing jobs on the DDP when the raw diagnostic data becomes available after a shot. Templates may be archived and used at a later date to rerun processing. In addition to shot related processing (this processing is initiated only during a timing sequence for a shot), operators may define 'immediate' processing. Immediate processing is scheduled to run as soon as the Process Data task has all data inputs available.

## Display of Results

Operators can request to view processing results from workstations by calling up a results list worksheet (Figure 7).

The results list worksheet contains names of results and their current status. Users can request to see

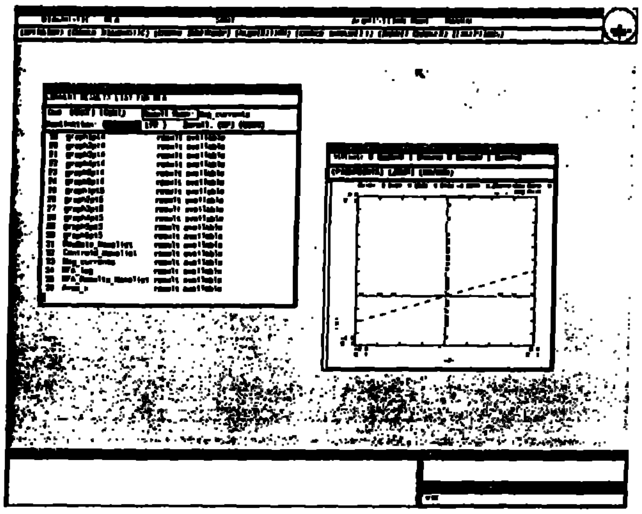

Figure 7. Results List Worksheet and a Result

lists of results for the latest shot, for an old (archived) shot, or for immediate processing. If a result name is available, an actual result display is requested by entering the name in the worksheet command banner, selecting the destination (a desktop window or the lineprinter), and for results from the latest shot, whether to display the result once or continuously. Continuous results display the named result for the most current shot--i.e. when a new shot goes off, the result display is automatically updated for the latest result with that name. The results list worksheet for the latest shot is updated every 15 seconds, so users can see when a particular result name actually becomes available (or if acquisition or processing has failed and the result will never be available).

Requesting a result from the results list worksheet causes a new process (a results display window) to be spawned from the results list worksheet task. This new results display process requests the result from the DDP 'Do Displays' task and creates the actual results display (Figure 8). Once a graphical result appears in a window, an operator may "zoom" in (magnify) parts of the graph, or may also pick a point on the graph (the coordinates of that point are displayed in the window banner).

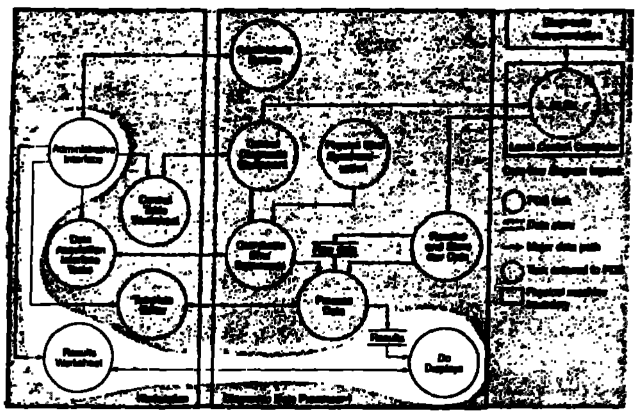

Figure 8. Data flow for viewing a result

Do Displays (DD) on the DDP is the clearinghouse for results. There is a separate results list, which is monitored by DD, for each diagnostic. When a completed result is requested by a workstation, or a previously requested result becomes available, DD transmits it to the proper window on the workstation.

## Conclusion

We have implemented the PDS user interface based upon the 'electronic desktop' metaphor. Our first diagnostic (MFA) has been implemented and successfully controlled from the desktop; processing results lists and actual results have been displayed in results worksheets on the desktop. The processing command and acquisition table worksheets have not as yet been integrated into the system, so at present processing definitions and acquisition lists are entered directly from the DDP. With the assistance of the PDS general-purpose utilities, installation of these remaining worksheets should prove straightforward.

There are several areas in which we would like to see continued development. While installation of worksheets is almost trivial, the development of the internal worksheet processes can be time-consuming. We would like to develop utilities or general-purpose tasks which will streamline this process. Also, handling of exceptions and alarms is still ambiguous and needs further consideration.

In general, initial feedback from actual desktop users seems to indicate that the current desktop is, as advertised, easy-to-use and provides useful and powerful tools. We believe that our design, based on the desktop metaphor, will facilitate future embellishments and continued utility.

## Acknowledgement

The authors are very grateful to their colleague Art Kobayashi for his time and assistance with this paper. We also wish to thank Ralph Jackson for his suggestions.

## References

[ 1] Butner, D.N., "MFTF Supervisory Control and Diagnostics System Hardware", PEPFR 8, pp. 2011-2012, November, 1979.

[ 2] Wyman, R.H., et al., "An Overview of the Data Acquisition and Control System for Plasma Diagnostics on MFTF-B", PEPFR 10, pp. 370-374, December, 1983.

[ 3] Smith, D.C., Irby, C., Kimball, R., Verplank, B., Harslem, E., "Designing the Star User Interface", Byte, 7(4) April 1982, pp. 242-282.

[ 4] Williams, G., "The Apple Macintosh Computer", Byte, 9(2) February, 1984, pp. 30-54.

[ 5] Ritchie, D.M., Thompson, K., "The Unix Time Sharing System", Comm. ACM 17, July 7, 1974, pp. 365-375.

[ 6] Kernighan, B.W., Ritchie, D.M., The C Programming Language, Prentice-Hall Inc., Englewood Cliffs, New Jersey, 1978.

[ 7] Sun Windows Programmer's Guide, Rev. A, 7 Jan 1984, Sun Microsystems, Inc., 2550 Garcia Ave., Mountain View, CA 94043.

[ 8] Preckshot, G.G., Goldner, A.A., Kobayashi, A., "Use of Spreadsheets for Interactive Control of MFTF-B Plasma Diagnostic Instruments", PEPFR 11, November, 1985.

[ 9] Deadrick, F.J., "Design of a Magnetic Field Alignment Diagnostic for the MFTF-B Magnet System", PEPFR 11, November, 1985.

[10] Labiak, W.G., "Software for the Local Control and Instrumentation System for MFTF", PEPFR 8, pp. 1991-1994, November, 1979.

[11] Minor, E.G., Labiak, W.G., "Tools and Methods for Implementing the Control System", PEPFR 9, pp. 2043-2045, October, 1981.

[12] Jackson, R.J., Balch, T.R., Preckshot, G.G., "Data Triggered Data Processing at MFTF-B", PEPFR 11, November, 1985.

[13] Preckshot, G.G., Saroyan, R.A., Mead, J.E., "A New Kind of User Interface for Controlling MFTF Diagnostics", PEPFR 10, pp. 358-361, December, 1983.

[14] McGoldrick, P.R., "Supervisory Control and Diagnostic System Distributed Operating System (SCDS)", PEPFR 8, pp. 2000-2002, November, 1979.

"Work was performed under the auspices of the U.S. Department of Energy by the Lawrence Livermore National Laboratory under contract number W-7405-ENG-48."